Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Please amend the claims as follows:

1. – 39. (Cancelled)

40. (Currently Amended) A method, comprising:

reading program code from memory and processing said program code with one or more processors to perform the following:

providing a user with options for modifying an application's bytecode, said application composed of a plurality of archive files, said archive files having respective class files, said respective class files having respective methods, said options including one or more of the following:

i) ~~modifying bytecode of classfiles within only one of said archive files;~~

[[i]]i) modifying bytecode of only one classfile within any one of said archive files;

[[i]]ii) modifying bytecode of only one method within [[any]]only one of said archive files' respective classfiles;

modifying bytecode of said application in accordance with said user's selection of one of said options;

executing said application in an object oriented runtime frame work, said executing including processing a portion of said application's bytecode that was modified in accordance with said user's selection of one or more of said options; and,

presenting to said user an output generated from execution of said portion of said application's bytecode that was modified.

41. (Currently Amended) The method of claim 40 wherein said object oriented runtime framework is a Java compliant object oriented runtime framework.

42. (Previously Presented) The method of claim 40 wherein said portion of said application's bytecode that was modified includes a method entry or method exit.

43. (Previously Presented) The method of claim 42 wherein said output is provided by a plug-in that a bytecode instruction inserted at said method entry or method exit is dispatched to.

44. (Previously Presented) The method of claim 43 wherein said output includes one of:

        a time at which said method entry or method exit was entered; and,

        a parameter that is passed at said method entry or method exit.

45. (Currently Amended) A machine readable storage medium containing program code that when processed by one or more processors of a computer causes a method to be performed:

providing a user with options for modifying an application's bytecode, said application composed of a plurality of archive files, said archive files having respective class files, said respective class files having respective methods, said options including one or more of the following:

        i) modifying bytecode of classfiles within only one of said archive files;

        [[i]]i) modifying bytecode of only one classfile within any one of said archive files;

        [[i]]ii) modifying bytecode of only one method within [[any]]only one of said archive files' respective classfiles;

modifying bytecode of said application in accordance with said user's selection of one of said options;

executing said application in an object oriented runtime frame work, said executing including processing a portion of said application's bytecode that was modified in accordance with said user's selection of one or more of said options; and,

presenting to said user an output generated from execution of said portion of said application's bytecode that was modified.

46. (Currently Amended) The machine readable storage medium of claim 45 wherein said object oriented runtime framework is a Java <u>compliant</u> object oriented runtime framework.

47. (Previously Presented) The machine readable storage medium of claim 45 wherein said portion of said application's bytecode that was modified includes a method entry or method exit.

48. (Previously Presented) The machine readable storage medium of claim 47 wherein said output is provided by a plug-in that a bytecode instruction inserted at said method entry or method exit is dispatched to.

49. (Previously Presented) The machine readable storage medium of claim 48 wherein said output includes one of:

a time at which said method entry or method exit was entered; and,

a parameter that is passed at said method entry or method exit.

50. (Currently Amended) A computer containing program code stored in memory of said computer that when processed by one or more processors of said computer causes a method to be performed:

providing a user with options for modifying an application's bytecode, said application composed of a plurality of archive files, said archive files having respective class files, said respective class files having respective methods, said options including one or more of the following:

~~i) modifying bytecode of classfiles within only one of said archive files;~~

[[i]]i) modifying bytecode of only one classfile within any one of said archive files;

[[i]]ii) modifying bytecode of only one method within [[any]]only one of said archive files' respective classfiles;

modifying bytecode of said application in accordance with said user's selection of one of said options;

executing said application in an object oriented runtime frame work, said executing including processing a portion of said application's bytecode that was modified in accordance with said user's selection of one or more of said options; and,

presenting to said user an output generated from execution of said portion of said application's bytecode that was modified.

51. (Currently Amended) The computer of claim 50 wherein said object oriented runtime framework is a Java compliant object oriented runtime framework.

52. (Previously Presented) The computer of claim 50 wherein said portion of said application's bytecode that was modified includes a method entry or method exit.

53. (Previously Presented) The computer of claim 52 wherein said output is provided by a plug-in that a bytecode instruction inserted at said method entry or method exit is dispatched to.

54. (Previously Presented) The computer of claim 53 wherein said output includes one of:

a time at which said method entry or method exit was entered; and,

a parameter that is passed at said method entry or method exit.